COURSE DESCRIPTION CARD - SYLLABUS

Course name

Programming Internet Applications

Course

Field of study	Year/Semester
Computing	1/1
Area of study (specialization)	Profile of study
Advanced Internet Technologies	general academic
Level of study	Course offered in
Second-cycle studies	Polish
Form of study	Requirements
part-time	compulsory

Number of hours

Lecture	Laboratory classes
30	30

Number of credit points 5

Lecturers

Responsible for the course/lecturer:
Marcin Borowski, BEng, PhD
email: mborowski@cs.put.poznan.pl
phone: 61 665 3032
faculty: Faculty of Computing and Telecommunications
address: Piotrowo 2, 60-965 Poznań

Responsible for the course/lecturer:
Marcin Borowski, BEng, PhD
email: mborowski@cs.put.poznan.pl
phone: +48 61 665 30 32
faculty: Faculty of Computing and Telecommunications
address: Piotrowo 2, 60-965 Poznań

Prerequisites

The student starting this course should have basic knowledge of structured and object-oriented programming, programming using the MVC scheme, basic knowledge of internet technologies (HTML, CSS, JS), and basic knowledge of database design.

Should have the ability to solve basic problems related to the process of designing IT systems and the ability to obtain information from the indicated sources.

Should also understand the need to expand their competences / be ready to cooperate within the team. Moreover, in terms of social competences, the student must present such attitudes as honesty, responsibility, perseverance, cognitive curiosity, creativity, personal culture, respect for other people.

Course objective

1. Provide students with basic knowledge on the design of web and mobile applications, the flow of information in such applications, technologies used in their construction (including applications for mobile devices).

2. Developing students' skills in solving problems related to the design of web and mobile applications, the use of Open Source solutions, frameworks, and libraries supporting the construction of such solutions.

3. Shaping students' teamwork skills and independence in solving problems.

Course-related learning outcomes

Knowledge

Student:

- has an orderly, theoretically founded knowledge of network technologies and internet applications related to the construction of internet applications

- has detailed knowledge related to selected issues in the field of computer science used in building web applications

- has knowledge of development trends in IT and in selected related disciplines - technologies used to build internet applications

- knows the life cycle of web applications, the life cycle of data processed in web applications

- knows advanced methods and techniques used in the implementation of engineering tasks related to building web applications

Skills

Student:

- can, when formulating and solving engineering tasks, integrate knowledge from various areas of computer science (and, if necessary, knowledge from other scientific disciplines) as well as knowledge in the area of operation of a given internet application and apply a system approach, also taking into account non-technical aspects

- can obtain information on Internet applications from literature, databases, and other sources (in Polish and English), integrate them and interpret and evaluate them, draw conclusions and formulate opinions

- can assess the usefulness and the possibility of using new technological achievements (methods, tools, libraries, frameworks, services) and new IT products

- can determine the directions of further learning and gain new knowledge about web applications and pass it on to others

Social competences
Student:

- understands the need to use the latest technology achievements and knows examples and understands the causes of malfunctioning applications that may lead to serious financial, image, or social losses

- understands that in computer science, knowledge and skills very quickly become obsolete, especially internet and mobile technologies

- can set priorities for the implementation of tasks defined by himself or others, in particular during the implementation of IT projects

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Formative assessment

a. Lecture: based on activity during the interactive parts of the lectures;

b. laboratory: based on the assessment of the current progress in the implementation of tasks;

Summative assessment

a. Lecture:

- assessment of the acquired knowledge and skills shown in the exam - a test conducted using an internet application; general and problem-related questions (the student can use any teaching materials; a selection test with about 30 questions; pass according to the following criteria of the obtained result:

  > 50% - 3.0
  > 60% - 3.5
  > 70% - 4.0
  > 80% - 4.5
  > 90% - 5.0

- discussion of the exam results

b. Laboratory:

- verification of the assumed learning outcomes realized by

- students' assessment and defense of the prepared tasks - 5 small projects;

When assigning the final grade, the student may obtain an increase in grade for:

• discussing additional aspects of the presented issues, not presented during classes;

• using skills and knowledge from outside the study program to solve the tasks performed;

• help in improving teaching materials related to the subject;

Programme content

Lecture:

The lecture program covers the following topics: The structure and syntax of HTML5, CSS3, and JavaScript. What's new in JavaScript in the context of HTML5. Structural and object-oriented programming in PHP. Overview of the Yii framework and creating applications based on this framework. Basics of node.js, express.js.

Laboratory:

Laboratory classes are conducted in the form of sixteen one-hour classes in the computer laboratory. The first classes are intended to familiarize students with the rules of using the laboratory and completing the exercises. The laboratory program covers the following topics: HTML5 website development. Creating CSS3, LESS, SASS style sheets. Basics of JavaScript language and supporting libraries. Programming in PHP. Building an application using the Yii framework. Creating simple applications in modern front-end technologies.

Teaching methods

Lecture: multimedia presentation, illustrated with examples given on the board.

Laboratory exercises: multimedia presentation, presentation illustrated with examples given on the whiteboard, live coding, and carrying out the tasks given by the teacher - practical exercises.

Bibliography

Basic

1. Technical documentation of the mentioned tools available on the Internet

2. jQuery Documentation - http://api.jquery.com

3. Yii2 Framework - http://www.yiiframework.com

4. Boostrap Framework – http://getbootstrap.com

5. ReactJS – https://pl.reactjs.org/

6. Angular – https://angular.io/

7. Svelte–https://svelte.dev/

Additional

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4
Classes requiring direct contact with the teacher	32 (16l, 16e)	1.2
Student's work (literature studies; preparation for laboratory classes; participation in consultations; preparation of programs, commissioning and testing; preparation for a final test)	68	2.8